

ABSTRACT OF THE DISCLOSURE

[1086] The design of nonblocking linked data structures using single-location synchronization primitives such as compare-and-swap (CAS) is a complex affair that often requires severe restrictions on the way pointers are used. One way to address this problem is to provide stronger synchronization operations, for example, ones that atomically modify one memory location while simultaneously verifying the contents of others. We provide a simple and highly efficient nonblocking implementation of such an operation: an atomic k-word-compare single-swap operation (KCSS). Our implementation is obstruction-free. As a result, it is highly efficient in the uncontended case and relies on contention management mechanisms in the contended cases. It allows linked data structure manipulation without the complexity and restrictions of other solutions. Additionally, as a building block of some implementations of our techniques, we have developed the first nonblocking software implementation of load-linked/store-conditional that does not severely restrict word size.